

ALGORITMA GENETIK PADA MASALAH TATA LETAK MESIN DENGAN PENGKODEAN KROMOSOM UNTUK UKURAN MESIN YANG BERBEDA-BEDA

Nelly Indriani Widiastuti

Program Magister Teknik Elektro, Institut Teknologi Bandung

E-mail: alifahth@yahoo.com

Abstract

The layout of the factory has a huge influence on the smooth production and production costs. One of the important issues in designing the layout of the factory is set up the layout of the engine. In this study used a genetic algorithm to determine the layout of the machine in a factory to obtain the value of the optimal material handling costs. This study uses a chromosome coding for the size of the different machines. This study solving problems at the factory layout engine which has 15 machines.

Keywords: *cost function, layout engine, genetic algorithm, chromosome encoding, the size of the machine*

Abstrak

Tata letak pabrik memiliki pengaruh yang sangat besar terhadap kelancaran produksi dan biaya produksi. Salah satu hal penting dalam merancang tata letak pabrik adalah mengatur tata letak mesin. Dalam penelitian ini digunakan algoritma genetik untuk menentukan layout mesin dalam sebuah pabrik untuk memperoleh nilai biaya penanganan material yang optimal. Penelitian ini menggunakan pengkodean kromosom untuk ukuran mesin yang berbeda-beda. Penelitian ini menyelesaikan masalah layout mesin pada pabrik yang memiliki 15 mesin.

Kata Kunci: *cost function, layout mesin, genetic algorithm, pengkodean kromosom, ukuran mesin*

I. PENDAHULUAN

Tata letak pabrik memiliki pengaruh yang sangat besar terhadap kelancaran produksi dan biaya produksi. Salah satu hal penting dalam merancang tata letak pabrik adalah mengatur tata letak mesin. Rancangan yang akan dibuat harus sesuai dengan disain fasilitas yang dalam hal ini adalah perancangan pada lantai, atau pengaturan penempatan peralatan fisik (peralatan, tempat, gedung, utilities). Tujuan rancangan tersebut adalah untuk mengoptimalkan hubungan antara bagian operasi, aliran material, informasi aliran untuk memperoleh hasil yang ekonomis, efektif, dan aman.

Masalah penanganan material sangat berkaitan dengan bagaimana mesin-mesin di lantai produksi diletakkan dengan cara yang tepat. *Layout* mesin yang baik akan menentukan biaya penanganan material optimal. Algoritma yang sering digunakan untuk menyelesaikan masalah ini sering tidak mempertimbangkan kondisi nyata ukuran mesin yang berbeda-beda dan kemungkinan posisi mesin yang cukup banyak. Dengan demikian perhitungan jarak antara mesin terbatas pada lokasi tertentu saja.

II. DASAR TEORI

Tujuan utama dalam desain tata letak pabrik pada dasarnya adalah meminimalkan total biaya yang antara lain meliputi elemen-elemen biaya sebagai berikut

- Biaya konstruksi dan instalasi fasilitas produksi
- Biaya pemindahan material (*material handling cost*)
- Biaya produksi, *maintenance cost*, *safety cost* dan biaya penyimpanan produk setengah jadi (*inventory in-process costs*)^[1]

Satu dari sekian banyak faktor yang dipertimbangkan dalam merancang *manufacture facility* adalah menemukan *layout* yang efektif. Definisi umum dalam masalah *layout* pabrik adalah menemukan pengaturan fisik fasilitas yang terbaik

untuk menghasilkan operasi yang efektif. Tata letak mempengaruhi biaya penanganan *material*, waktu dan *throughput*, dan akhirnya mempengaruhi produktifitas secara keseluruhan dan efisiensi pabrik^[1].

Berdasarkan algoritma yang berbeda, masalah layout dapat memiliki model yang beda. Menurut fungsi tujuan, ada dua tujuan dasar yaitu, meminimalkan total biaya penanganan *material* dan tujuan lain adalah memaksimalkan nilai jarak^[1]. Formulasi model untuk meminimalkan total biaya penanganan *material* adalah sebagai berikut:

$$\min z = \sum_{i=1}^m \sum_{j=1}^n f_{ij} c_{ij} d_{ij} \quad (1)$$

Dimana,

Z = total biaya penanganan *material*

f_{ij} = alur *material* dari mesin i ke mesin j

c_{ij} = biaya untuk memindahkan satu unit muatan per satu unit jarak antara dua mesin

d_{ij} = jarak antara mesin i dan j

m = jumlah mesin

Dengan mengasumsikan biaya c_{ij} konstan, tujuan akan berubah untuk meminimalkan total jarak yang dilalui semua *parts*.

Genetik algoritma

Algoritma genetika adalah algoritma pencarian heuristik yang didasarkan atas mekanisme evolusi biologis. Keberagaman pada evolusi biologis adalah variasi dari kromosom antar individu organisme. Variasi kromosom ini akan mempengaruhi laju reproduksi dan tingkat kemampuan organisme untuk tetap hidup. Algoritma ini menggunakan metoda untuk bergerak dari sebuah populasi kromosom menjadi populasi yang baru menggunakan sejenis seleksi alami dan operator genetik, *crossover*, *mutation* dan *inversion*. Tiap kromosom berisi gen, misalnya bit, tiap gen menjadi *allele* tertentu.

Pada setiap generasi, kromosom akan melalui proses evaluasi dengan menggunakan alat ukur yang disebut dengan fungsi *fitness*. Operator seleksi

memilih kromosom tersebut dalam populasi yang akan diijinkan untuk bereproduksi.

Algoritma genetik memiliki 6 komponen utama, yaitu :

1. teknik pengkodean,
2. prosedur inisialisasi,
3. fungsi evaluasi,
4. seleksi,
5. operator genetik,
6. penentuan parameter^[6].

1. Teknik pengkodean

Pengkodean harus merepresentasikan individu yang akan dicari solusinya. Operator genetik harus dapat diterapkan pada sandi-sandi tersebut. Gen dapat direpresentasikan dalam bentuk : string bit, *tree* atau pohon, array bilangan real, daftar aturan, elemen permutasi, elemen program dan lain-lain.

2. Prosedur inisialisasi

Prosedur inisialisasi adalah tahapan menentukan populasi awal. Proses ini membangkitkan sejumlah individu secara acak atau melalui prosedur tertentu. Ukuran populasi tergantung pada masalah yang akan dipecahkan dan jenis operator genetik yang akan diimplementasikan.

3. Fungsi evaluasi

Dalam fungsi evaluasi ada dua hal yang harus dilakukan dalam mengevaluasi kromosom yaitu : evaluasi fungsi objektif dan konvergensi fungsi objektif ke dalam fungsi *fitness*.

Fungsi objektif selanjutnya akan menjadi acuan bagi nilai *fitness*. Nilai *fitness* adalah nilai yang menyatakan baik tidaknya sebuah individu. Penentuan nilai *fitness* tergantung pada tujuan permasalahan yang akan diselesaikan. Nilai *fitness* dapat diminimalkan atau dimaksimalkan.

4. Seleksi

Seleksi bertujuan untuk memberikan kesempatan reproduksi yang lebih besar bagi anggota populasi yang paling fit. Dalam seleksi akan ditentukan individu-individu mana yang akan dipilih untuk dilakukan rekombinasi dan bagaimana

offspring terbentuk dari individu tersebut. Yang paling umum dilakukan adalah crossover dan mutasi.

5. Operator genetik

Operator genetika adalah metoda yang digunakan dalam algoritma genetik untuk melakukan perubahan terhadap sebuah kromosom untuk menghasilkan individu yang baru.

6. Penentuan parameter

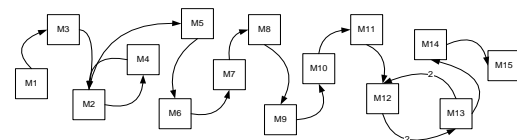
Parameter adalah parameter kontrol algoritma genetika, yaitu : ukuran populasi (*popsi*), peluang *crossover* (pc) dan peluang mutasi (pm)

III. RANCANGAN SOLUSI

Dalam masalah pengaturan tata letak mesin pabrik banyak variabel yang harus dipertimbangkan.

Asumsi yang dibuat untuk penelitian ini adalah sebagai berikut:

1. Jumlah mesin yang digunakan ada 15 buah
2. Bagian produk yang akan dibuat adalah satu bagian yaitu, chasis engine. Produk tersebut dikerjakan satu kali.
3. Komponen frekuensi dapat dilihat pada gambar 1. Beserta alur produksi.
4. Jarak sebagai komponen penghitungan nilai cost adalah koordinat titik pusat mesin.
5. Alat yang digunakan untuk mengangkut material diasumsikan sejenis, sehingga komponen biaya MHD dianggap satu satuan.
6. Ukuran mesin berbeda-beda. Hal ini akan mempengaruhi komponen jarak.



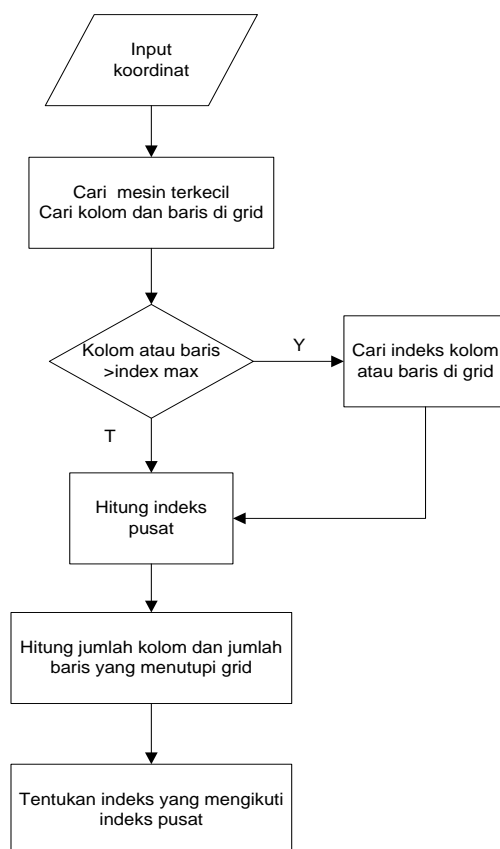
Gambar 1.alur produksi 1 part

Representasi lantai pabrik

Dalam tesis ini kromosom dikodekan dengan menjadikan ukuran mesin sebagai gen. Dengan algoritma genetik ini, lantai pabrik yang dibuat menjadi grid yang akan menjadi area solusi bagi posisi mesin.

12	13	3.5	24.96	6.72
13	8	4	15.36	7.68
14	2	1.5	3.84	2.88
15	2	1.5	3.84	2.88

Berikut ini adalah diagram alur yang menggambarkan proses penterjemahan koordinat mesin yang diperoleh dari hasil penataan mesin yang dilakukan oleh user pada aplikasi MflaSh menjadi indeks lokasi lantai pabrik.



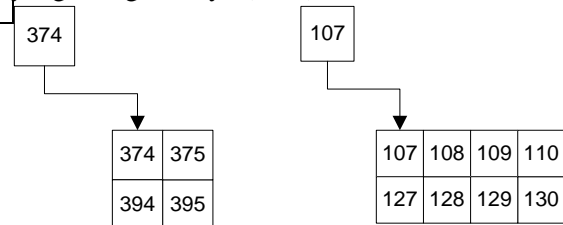
Gambar 4. Diagram alur konversi koordinat mesin menjadi indeks mesin

Indeks yang telah didapat dari hasil konversi tersebut akan digunakan sebagai representasi parameter algoritma genetik yang akan diacak dan dikembangkan menjadi individu baru sesuai dengan nilai *fitness*.

Dengan demikian *string* kromosom mesin adalah sebagai berikut :

**{1188;441;626;195;1578;328;623;450;76
2;204;426;1403;84;623;768}**

Setiap gen dalam kromosom memiliki indeks yang mengikuti. Dibawah ini adalah ilustrasi indeks pusat (*idxC*) dan indeks yang mengikutinya (*idxF*).



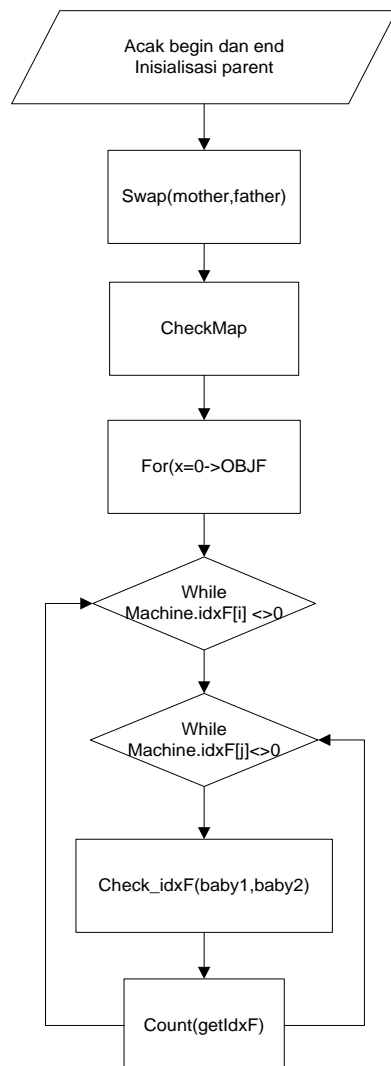
Gambar 5. Ilustrasi contoh struktur gen pada kromosom

Crossover

Tujuan operator *crossover* adalah membuat keturunan (*offsprings*) baru dengan melakukan kombinasi dan mengatur ulang kembali bagian dari individu orangtuanya. Dalam penelitian ini metoda *crossover* yang digunakan adalah *Partial Mapped Crossover* (PMX). Frekuensi *crossover* dikontrol oleh parameter yang disebut probabilitas *crossover* (p_c). Dalam metoda ini ditentukan dua titik acak sebagai posisi penukaran (*begin* dan *end*) yang berada diantara 1 sampai $n-1$.

Setelah menentukan pasangan parent, kedua titik tersebut mulai dari *gen[begin]* sampai *gen[end]* ditukarkan antara parent tersebut. Setelah proses penukaran, *idxC* dan *idxF* seluruh mesin dihitung untuk memastikan tidak ada duplikasi pada gen yang merupakan indeks pusat (*idxC*) mesin maupun indeks yang mengikutinya (*idxF*) mesin lain.

Pada gambar 6 adalah algoritma operasi *crossover* PMX yang disertai dengan mekanisme yang memastikan bahwa tidak ada indeks pusat yang sebuah individu menempati indeks milik individu lain. Cara yang digunakan untuk memastikan bahwa tidak ada indeks yang tumpang tindih adalah mekanisme mencocokkan total jumlah indeks setiap mesin (*totIdx*) dengan jumlah indeks yang diperoleh setelah *crossover* (*getIdx*).



Gambar 6. Diagram alur crossover PMX dan cek indeks.

Alur Material

Aliran *material* untuk melakukan part 1, dapat dilihat pada Gambar 1. Aliran material tersebut digunakan untuk menghasilkan biaya penanganan material yang digunakan sebagai nilai *fitness*. Parameter yang menentukan biaya penanganan *material* adalah :

- 1) Jarak dari mesin satu ke mesin berikutnya
- 2) Frekuensi kerja masing-masing mesin untuk pengerjaan part 1

Rumus biaya penanganan material dapat dilihat pada persamaan (1). Untuk kebutuhan persamaan tersebut, parameter frekuensi ditentukan sebagai ketetapan. Parameter jarak diperoleh dari hasil

konversi indeks yang dimiliki masing-masing mesin menjadi koordinat mesin. Selanjutnya dari koordinat tersebut dapat dihitung jarak antar mesin dengan persamaan berikut :

$$D = \sqrt{|x_1 - x_2|^2 + |y_1 - y_2|^2} \dots (2)$$

Dimana (x_1, y_1) adalah koordinat mesin satu dan (x_2, y_2) adalah koordinat mesin berikutnya sesuai urutan graph produksi.

Fungsi tujuan

Nilai *fitness* akan menentukan individu yang layak untuk dipertahankan dan dikembangkan. Dalam penelitian ini nilai *fitness* digunakan untuk menentukan layout mesin yang akan diseleksi dan direkombinasi dengan layout mesin yang lain. Hal ini untuk memperoleh kemungkinan nilai *fitness* yang lebih baik.

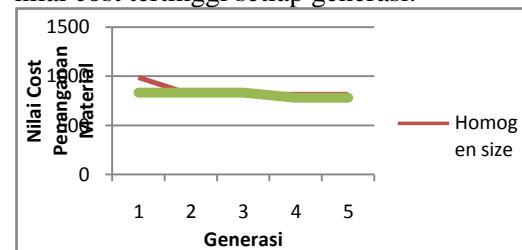
Fungsi objektif yang digunakan adalah persamaan (1), nilai yang menjadi tujuan adalah nilai minimal *cost* penanganan material. Dengan demikian nilai *fitness* yang dipilih adalah nilai *fitness* yang maksimal.

Berikut ini adalah nilai *fitness* yang digunakan dalam penelitian ini.

$$\max fitness = \frac{1}{\sum_{i=1}^m \sum_{j=1}^n f_{ij} c_{ij} d_{ij}} \quad (3)$$

IV. HASIL PENELITIAN

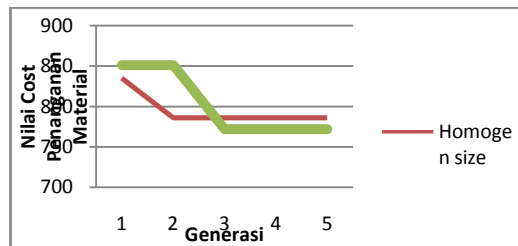
Percobaan dilakukan pada beberapa set parameter data. Berikut adalah beberapa hasil percobaan. Grafik menggambarkan nilai *cost* tertinggi setiap generasi.



Gambar 7. Grafik perbandingan nilai optimal untuk jumlah populasi 24

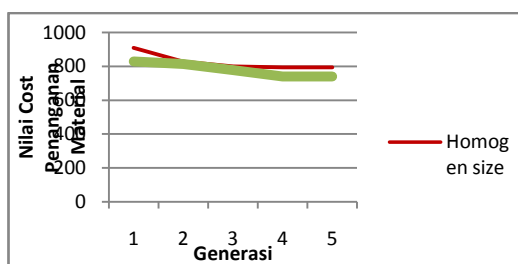
Hasil yang diberikan *Heterogenous_Size* lebih baik dari *Homogenous_Size* secara keseluruhan.

Kurva yang dibentuk Heterogenous_Size mempunyai bentuk yang hampir sama dalam setiap percobaan. Pada setiap percobaan Heterogenous_Size menghasilkan nilai cost yang lebih rendah dibanding dengan nilai cost yang dihasilkan Homogenous_Size.



Gambar 8. Grafik perbandingan nilai optimal untuk jumlah populasi 16.

Secara umum percobaan dengan jumlah populasi kurang dari 30 dan peluang crossover 0,90, dengan jumlah generasi 5, membentuk kurva yang lebih landai atau grafik menunjukan konvergen pada generasi ke-3, parameter baik dengan populasi 24 maupun 16. Pada percobaan dengan jumlah jumlah generasi 10, grafik menunjukan nilai optimal pada generasi ke-4 pada percobaan dengan jumlah populasi 24 dan optimal pada generasi ke-2 dengan jumlah populasi 16.



Gambar 9. Grafik perbandingan nilai optimal untuk jumlah populasi 32.

Pada percobaan dengan jumlah populasi lebih dari 30 dan peluang crossover 0,70, menunjukan konvergen yang lebih lama baik pada jumlah generasi 5 maupun dengan jumlah generasi 10.

V. KESIMPULAN

Kesimpulan yang dapat diambil pada penelitian tesis ini, berdasarkan dalam

hasil percobaan-percobaan seperti di bawah ini.

1. Penelitian ini berhasil membuktikan bahwa algoritma genetik dengan pengkodean kromosom untuk ukuran mesin yang berbeda-beda, menghasilkan nilai cost penanganan material yang optimal.
2. Hasil penelitian yang diperoleh mendukung dan sesuai dengan sifat algoritma genetik. Hasil yang diperoleh merupakan pembuktian berbentuk simulatik.
3. Dalam melakukan percobaan, pemilihan parameter algoritma genetik dilakukan sesuai dengan landasan teori dan dilakukan berulang-ulang untuk hasil yang baik.
4. Jika jumlah populasi tinggi, maka individu-individu melakukan kombinasi dan rekombinasi dengan operator crossover, menjadi lebih banyak. Hal tersebut menyebabkan algoritma genetik menemukan nilai cost yang lebih bervariasi.
5. Algoritma genetik dengan pengkodean kromosom untuk ukuran mesin yang berbeda-beda, menunjukan hasil yang lebih konvergen, sehingga diperoleh nilai cost penanganan material yang lebih cepat.
6. Pengkodean kromosom untuk ukuran mesin yang berbeda-beda, menyebabkan penentuan nilai optimal yaitu jarak mesin menjadi lebih dekat, sehingga dapat lebih cepat menemukan nilai optimal.

VI. DAFTAR PUSTAKA

- [1] Apple, James M, (1977), *Plant Layout and Material Handling*, John Wiley & Sons, Third Edition
- [2] David E. Goldberg, (1989), *Genetic Algorithms in Search, Optimization & Machine*

Learning, Addison-Wesley Publishing Company

- [3] Randy L. Haupt, Sue Ellen Haupt (1998), *Practical Genetic Algorithms*, John Wiley & Sons.
- [4] Hassan, MMD., 1995, *Layout design in group technology manufacturing*, *International Journal of Production Economics*, Vol. 38, Part 2-3, pp173-188, http://econpapers.repec.org/article/eeeproeco/v_3a38_3ay_3a1995_3ai_3a2-3_3ap_3a173-188.htm, 30/2/2009, 13:29 WIB
- [5] Morad, Norhashimah, *Genetic Algorithms Optimization For The Machine Layout Problem*, School of Industrial Technology, Universiti Sains Malaysia, Penang, Malaysia, e-mail: nhashima@usm.my, <http://www.journal.au.edu/ijcim/jan00/morad.doc>, diakses Kamis 26 Februari 2009, 09:41 WIB.
- [6] Yang, Taho, *Systematic layout planning : a study on semiconductor wafer fabrication facilities*, National Checnng Kung University, Tainan, Taiwan.
- [7] _____, *Algoritma Genetika*, <http://ocw.gunadarma.ac.id/course/industrial-technology/informatics-engineering-s1/pengantar-kecerdasan-buatan/algoritma-genetika>, 24/3/2009, 08:39 WIB.